

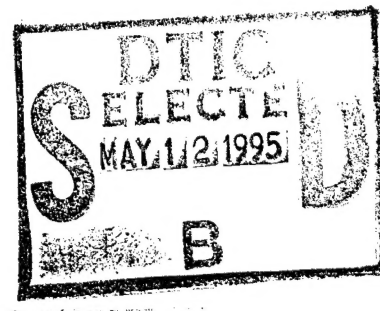
TOWARDS FRIENDLIER USER INTERFACES FOR EXPERT SYSTEMS

Elaine Marsh

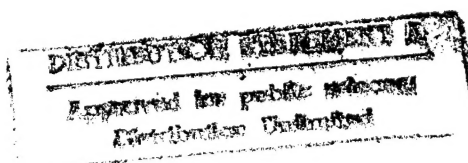
Naval Research Laboratory, Washington, DC, USA

ABSTRACT

Much research in computational linguistics and artificial intelligence has gone into characterizing natural language discourse structures, developing discourse models, and using these to develop more useful natural language interfaces to databases and expert systems. However, this knowledge about the nature of discourse has not been used to any advantage in graphical expert system interfaces. This paper proposes that discourse elements and discourse models used in the world of natural language understanding can be extended to graphical expert system interfaces, promoting continuity of interactions and allowing relationships among interactions to be exploited, redundancy reduced, operations tracked, and higher-level operations recognized.



19950510 122



DTIC STATEMENT 19950510 122

THE PROBLEM

Traditional graphical interfaces to expert systems are designed to allow direct manipulation of objects. Traditional interfaces to expert systems have been command-oriented, augmented with menus, mice, and other means to provide direct manipulation. The type of manipulation is typically restricted and quite simple. For example, manipulations may include walking menus, dragged folders, or mouse actions. An ordered set of manipulations is frequently required to perform a given operation. It can require a high degree of familiarity with the system interface for the user to perform the set of manipulations in the prescribed sequence. As an example case, we will look at the interface to TABS, an anti-submarine warfare battle management system under development at the Naval Research Laboratory. Table 1 contains an example of a dialog with TABS. The manipulations required to use the graphical interface are given on the left; a set of what we might expect in a natural language dialog with the system are given on the right.

There are several marked differences between the interface interactions on the left and on the right.

- In the graphical system, there is little relationship between the manipulations or between the operations. Individual menu-driven commands are concise. Walking menus do provide some continuity and dependency relationships among manipulations; however, in general, higher level operations are distinct and unconnected. In the natural language dialog, there are explicit and implicit relationships between interactions, reflecting the user's understanding of the system, its operation, and the course of the dialog. For example, in utterance (D), the pronoun "IT" refers to the Oscar of the previous utterance (C) and not the Victor-III of the real-world situation in (A). A natural language interface which understands discourse elements and which has a way of representing a model of the discourse allows continuity and connectedness between the interactions.
- When menus are used in graphical interfaces, all commands must be entered in sequence; natural language interactions allow telescoping of information through the use of conjunction, ellipsis, and pronominal referencing. For example, in utterance (H), the use of the conjunction "AND" allows two or more assertions to be combined into one sentence, telescoping entire series of commands into one string.
- Graphical interfaces do not typically allow previous operations to be referred to explicitly or implicitly, nor can they provide higher level explanations of operations. Graphical interfaces can track individual manipulations via history lists, but only at a very low level. Natural language interfaces with discourse capabilities use a discourse model to describe the order and relationships between interactions. Change of focus, hypothetical worlds, causality, and other higher level objects can be represented. For example, a discourse model would need to detect that the world changes from a real one to a hypothetical one between utterances (A) and (C).
- The graphical interface requires that the user understand and follow an arbitrary order of manipulations. For example, in the sixth and seventh set of manipulations, related to utterances F and G, the user must repeat interactions (4-9), with the proper substitutions. Each manipulation must be performed in the required order, without deviation. So manipulations are frequently repetitive, redundant, and seemingly unconnected to previous or following manipulations. Natural language interfaces permit information in smaller, non-fragmented and less cryptic packages than menu-driven interactions, for example [Paap & Roske-Hofstrand 1988; Perry & Voelcker 1989], yet with them, the user can have a higher level picture of the operations within the context of the expert system. Thus in utterance (C), the user changes the focus of the dialog by specifying a new submarine class, but need not know the details of the implementation to perform the act. For example, the user would not need to know that operations 4-7 must be performed in a specific order.

In summary, natural language provides the user with a natural and simple means to interface to an expert system. Varied user populations are easily accommodated and, potentially, training time is reduced since users do not need to learn complex query or command languages or complex sets of commands in a menu-driven system. The number of redundant and repetitive interactions is reduced and more freedom is



Availability Codes	
Dist	Special
A-1	

TABLE 1
Examples of Expert System Interface Interactions

GRAPHICAL INTERFACE	ENGLISH
1. Click on Victor-III. 2. Pull down on "Layercake" menu to "Display".	<i>A. Tell me about the Victor-III.</i>
3 Drag folder for 0700Z into "Viewed" window.	<i>B. What about at 0700Z?</i>
4. Drag folder from "Viewed" to "What-if" window. 5. Pull down on "LayerCake Menu" to select "Change Classification". 6. Pull down on "Poss. Sub. Classif." menu to "Oscar". 7. Drag "What-if" folder from "Viewed" window to "Forecast Tac Pic" window.	<i>C. What if it was an Oscar?</i>
8. Wait for forecast folders for +2.00, +4.00, +6.00, etc. to appear in "Viewed" window and move to "Forecast Tac Pic" window. 9. Drag "0700Z +2.00" folder to "Viewed" window.	<i>D. Where would it be at +2.00?</i>
10. Drag "0700Z +8.00" folder to "Viewed" window.	<i>E. At +8.00?</i>
<i>Repeat (4-9) and substitute "Charlie" for "Oscar" in (6).</i>	<i>F. What about a Charlie?</i>
<i>Repeat (4-9) and substitute "Delta" for "Oscar" in (6), and "+4.00" for "+2.00" in (9).</i>	<i>G. A Delta at +4.00?</i>
11. Allow real-time tactical picture update to occur. 12. Move "0900Z" from "Viewed" window to "Real-time Tac Pic Update" window. <i>Repeat (1-3).</i> 13. Pull down "LayerCake" menu to "Show3 in loc". 14. Pull down "LayerCake" menu to "Show threat radius".	<i>H.</i> <i>Show the Victor's current IL density grid and its threat radius.</i>

allowed in the order of operations. Continuity among interactions is promoted and the use of implicit information maximized.

Within the scope of natural language interfaces and natural language understanding systems in general, a substantial amount of research has gone into characterizing discourse structures and developing models of discourse and strategies for discourse processing. However, this wealth of knowledge about the nature of discourse has not been used to any advantage in other types of interfaces. It is the purpose of this paper to suggest that the techniques and knowledge about discourse developed for natural language systems can be exploited and extended to other types of man-machine interfaces.

In this paper, we will identify some of the implicit and explicit elements in natural language interactions that carry discourse information, describe the sorts of relationships that can hold, and show how this discourse analysis might be applied to graphical interfaces to expert systems to promote more natural

man-machine interactions. It is hoped that the results of our ongoing work will promote description, evaluation, and improvements in man-machine interfaces.

DISCOURSE MODELING OF NATURAL LANGUAGE DIALOGS

Let us consider how discourse relations among sets of sentences can be represented.¹ Discourse relations can be either local or global. Local connections are restricted in their scope and effect. For example, some referents can be determined from the most recently mentioned objects that meet selectional restrictions.² For local connections, keeping a history list is sufficient for keeping track of possibilities. For example, in Table 1 above, the focus of the dialog in (C, F, G), i.e. the introduction of different submarine classes, can be kept track of relatively easily in natural language systems that have simple reference resolution via history lists. However, in extended dialogs, where larger fragments of discourse must be handled, topics can shift over the course of the interactions. In these cases, history lists may not provide the solution to the problems. One solution might be to segment the dialog into context spaces [Reichman, 1985; Allen, 1987], where each space represents a stretch of discourse where sentences are addressing the same topic and the sequence of sentences display a local coherence. [Allen 1987] suggests several properties that should hold for a given discourse segment or context space: local relations should hold, including reference and ellipsis; a simple progression of time and location should run through the segment; there should be a fixed set of speakers and hearers and a fixed set of background assumptions.

Figure 1 is an illustration of how the natural language dialog from Table 1 can be partitioned into a set of context spaces. In each of the spaces, the context changes: topics are revised, time specifications

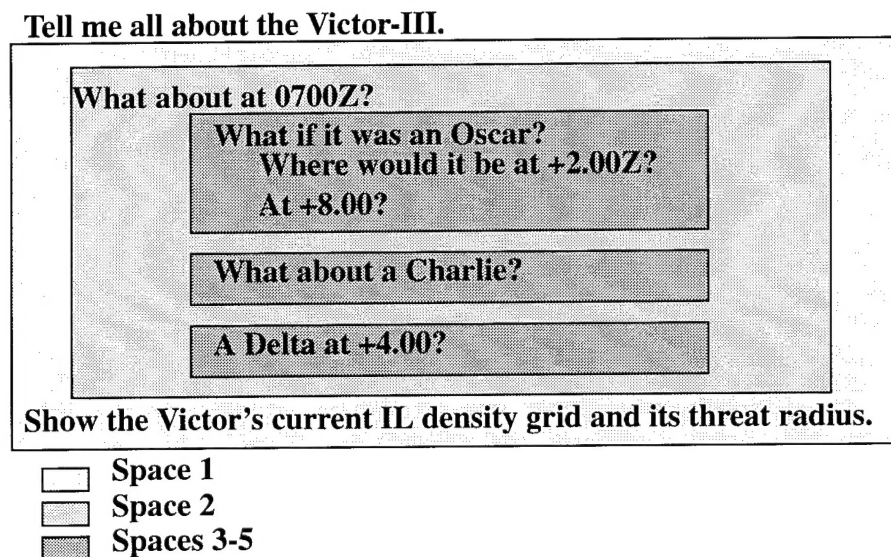


FIGURE 1. Context Spaces.

change, and additional assumptions are made. Space 1 contains the discourse about the Victor-III at the current time. In Space 2, the discourse jumps back to a previous time period, although the focus is the same. A new assumption is made in Space 3, with the introduction of a hypothetical world and a new focus introduced. However, we assume that some of the objects of concern in Spaces 1 and 2 might also be

1. Some definitions are provided here for non-computational linguists, so that the discussion can be more easily followed. A *referent* is an object that is being discussed and referred to. *Anaphora* is the process of finding a referent for a pronoun. *Ellipsis* is a syntactic phenomenon where certain elements are left out of a sentence, typically leaving a sentence fragment for which missing information must be identified. Focus is the particular object that must be referred to in the segment of the dialog.

2. Selectional restrictions eliminate semantically anomalous word senses by matching semantic patterns against syntactic structures. For example, a semantic pattern saying that *eat* verbs take animate subjects would rule out a sentence like *Rocks eat cake*.

carried through, e.g. location of the submarine, its surroundings, etc. In Spaces 4-5, new foci and time periods are discussed and the assumption of the hypothetical world is carried forward. Finally, we return to the original context space with a query about the current status of the Victor. Notice that the focus has shifted over the course of the dialog from the Victor, at the beginning of the dialog, to other submarine classes, and back to the Victor. Through the use of context spaces it becomes possible to recognize that the referent is the original Victor and not a new entity. Furthermore conditions are restored to the same conditions in effect at the beginning of the dialog.

It is important to recognize discourse features in the dialog in order to properly model the discourse and segment it into context spaces. Discourse information can be carried either explicitly or implicitly in the dialog, i.e. either there are explicit cues indicating discourse relations, or discourse relations will have to be inferred from other information present in the dialog. Let us look at the natural language interactions from our example dialog, repeated in the left column of Table 2. Associated discourse features are indicated in the right column.

TABLE 2
Examples of Expert System Interface Interactions

A. Tell me all about the Victor-III.	<i>User Model, Reference</i>
B. What about at 0700Z?	<i>Time, User Modeling</i>
C. What if it was an Oscar?	<i>Anaphora, Focus, Reference</i>
D. Where would it be at +2.00?	<i>Anaphora, Time</i>
E. At +8.00?	<i>Ellipsis, Time</i>
F. What about a Charlie?	<i>Ellipsis, Reference, Focus</i>
G. A Delta at +4.00	<i>Ellipsis, Time, Reference, Focus</i>
H. Show the Victor's current IL density grid and its threat radius.	<i>Reference, Anaphora, Focus, Conjunction</i>

Explicit cues to discourse relations are elements that are actually present in the text. They promote the flow of the dialog and signal changes in assumptions, time, and focus, among other things. Examples of discourse cues include elements such as *what if*, *therefore*, time prepositions, pronouns, definite articles on noun phrases, etc. Explicit discourse markers are specific syntactic and lexical elements in the dialog that indicate particular relationships. Sentential relationships may be hypothetical (*what if*, *suppose*), causal (*therefore*, *as a result*), temporal (*then*, *before*, *after*), or conjunctive (*and*, *or*, *but*), among others. Explicit discourse cues can also indicate other kinds of discourse relationships. Referential expressions such as pronouns (e.g. *it*, *its* in utterances (D) and (H)), and definite noun phrases (*the Victor* in utterance (H)), relate objects in the dialog. Explicit time expressions (such as *0700Z* in (B)) can relate sequences of events. To understand the dialog, such markers must be identified, represented, and understood [Fox 1990]. For example, in (C) the use of *what if* is an explicit syntactic cue that we are entering a hypothetical world. Objects and actions in the real world and hypothetical world must be tracked. Some objects may overlap between worlds; others may not. For example, while the submarine object changes to an *Oscar*, the rest of the scenario may change or remain constant. When the dialog moves back from the hypothetical world to the real world, there is a change of focus and objects in the hypothetical world disappear and objects from the real world return.

While explicit discourse markers are useful in modeling natural language dialogs, much of the discourse information is implicit and must be inferred from context. Discourse analysis should confirm these implicit discourse relations, such as causality and temporal order, through inferencing. For example, the time relations between sentences and paragraphs can be inferred and a temporal order of events and actions hypothesized [Allen 1984]. Discourse processing should be able to hypothesize what is to be

discussed, what types of extended interactions are possible in the course of the dialog, and what the appropriate actions are [Carberry 1988]. For example, in utterance (A) it should understand what sorts of things are required to provide all relevant information about the Victor. It should recognize and understand changes in topic and focus [Reichman 1985; Sidner 1983]. In utterances (C, F, G, H) the focus of the dialog changes as new submarine classes are requested. It should be able to use prior context to resolve NP reference (including pronoun resolution) and ellipsis [Carberry 1989; Hirst 1987] for utterances (B,C,D,E,F,G). While certain syntactic constructions are evident when information is missing, e.g. sentence fragments in cases of ellipsis (B, E, F, G), pronouns in the case of reference (C, F, G), much discourse information must be inferred from the structure of the discourse and knowledge of the world. For example, it must be recognized that the Victor mentioned in (H) is the same Victor as the one mentioned explicitly in utterance (A) and implicitly in (B). Even temporal relationships can be implicit in the dialog. For example, we are assuming that the example dialog has taken place relatively quickly and that (A) and (H) occur within the same general time frame, so *current* in (H) means the same time for both. If a more extended dialog had taken place, with additional interactions intervening, substantial time may have passed between the first request in (A) and the last (H). The implicit time progression must be recognized so that changes in the situation over time can be reflected in the response of the system, e.g. the location of the submarine may have changed and so its threat radii may have changed.

We have seen that we can model natural language dialogs by identifying discourse relations and segmenting the discourse into discourse segments. Explicit information and implicit information from the narrative are used in order to build up context spaces.³ In the next section, we consider how these notions might be applied to graphical interfaces.

DISCOURSE MODELING FOR GRAPHICAL INTERFACES

An interface that is capable of discourse analysis can identify relevant facts, store this information, and retrieve information based on those elements in the discourse, much like the human does when conversing with another human. This would eliminate much of the redundancy in interacting with a graphical interface. Discourse modeling can help in determining the appropriate action and appropriate level of response for the user since information about actions and the world can be retained in a rigorous way [Scha 1988; Reichman 1985; Grosz and Sidner 1986]. Finally, although there is currently no representation of discourse structure in expert systems and their graphical interfaces beyond keeping track of local relations among objects and actions, information about what is current and what is old could be made available. By modeling the discourse in graphical interfaces it becomes possible to recognize relationships among interactions and highlight the focus or topic of interest. For example, it might be possible to highlight changes in the situation or focus that have occurred the fact over the course of the dialog

In Table 1 above, the graphical interface has information available from user interactions with menu items (commands), cursor locations (text input), mouse actions (selecting, deselecting, dragging) and graphics (e.g. maps, icons, folders). Let us hypothesize how discourse information might be represented for a graphical interface and see whether these can be related to natural language discourse properties. In Table 3, we relate interactions from a graphical interface to the discourse elements we discussed above. We can relate many of the direct manipulations to discourse phenomena we identified in the natural language dialog. For example, mouse clicks are referential actions (1), just as pronouns or names are referential expressions. Selection of menu items and dragging of folders change our view about the current world, whether it is a change in focus (6(X3), 12, etc.), time (3, 9, 10, etc.) or assumptions (4). Graphical elements, like maps, icons, create a picture of the current situation and thus reflect a user model (2).

We believe that techniques used in natural language technology for discourse processing can be extended to graphical interfaces. Just as we can develop a discourse model for discourse segments in a natural language dialog, we can develop a discourse model for interactions with graphical interfaces.⁴

3. For ideas on how to implement a discourse component for natural language systems, see [Allen 1987].

4. Although we will not draw the discourse model for the graphical interface here, we hypothesize that it will look very much like the one presented in Figure 1.

TABLE 3
Discourse Elements for Graphical Interfaces

GRAPHICAL INTERFACE	DISCOURSE ELEMENT
1. Click on Victor-III. 2. Pull down on "Layercake" menu to "Display".	<i>reference</i> <i>user model, new action</i>
3 Drag folder for 0700Z into "Viewed" window.	<i>time</i>
4. Drag folder from "Viewed" to "What-if" window. 5. Pull down on "LayerCake Menu" to select "Change Classification". 6. Pull down on "Poss. Sub. Classif." menu to "Oscar". 7. Drag "What-if" folder from "Viewed" window to "Forecast Tac Pic" window.	<i>assumption revision</i> <i>new action</i> <i>focus</i> <i>new action</i>
8. Wait for forecast folders for +2.00, +4.00, +6.00, etc. to appear in "Viewed" window and move to "Forecast Tac Pic" window. 9. Drag "0700Z +2.00" folder to "Viewed" window.	<i>time progression</i> <i>time/reference</i>
10. Drag "0700Z +8.00" folder to "Viewed" window.	<i>time/reference</i>
Repeat (4-9) and substitute "Charlie" for "Oscar" in (6).	<i>reference/focus</i>
Repeat (4-9) and substitute "Delta" for "Oscar" in (6), and "+4.00" for "+2.00" in (9).	<i>reference/focus</i>
11. Allow real-time tactical picture update to occur. 12. Move "0900Z" from "Viewed" window to "Real-time Tac Pic Update" window. Repeat (1-3). 13. Pull down "LayerCake" menu to "Show3 in loc". 14. Pull down "LayerCake" menu to "Show threat radius".	<i>assumption revision</i> <i>time</i> <i>time</i> <i>focus</i> <i>new focus</i> <i>new focus</i>

Capability of graphical interfaces to carry on cohesive dialogs with users will be improved. Redundancy and repetitive manipulations will be reduced. Active objects will be more easily identified and kept track of, and discourse elements such as temporal relations and changes in focus will be more readily accessible. Dialog with expert system interfaces will become friendlier, less opaque, and more natural.

In the next section, we describe a standard architecture for natural language interface systems and describe where discourse analysis plays a part in hopes that expert system developers will incorporate discourse analyzers into their graphical interfaces.

AN ARCHITECTURE FOR DISCOURSE PROCESSING

Natural language interface systems (and natural language processing systems in general) characteristically have the architecture illustrated in Figure 2.

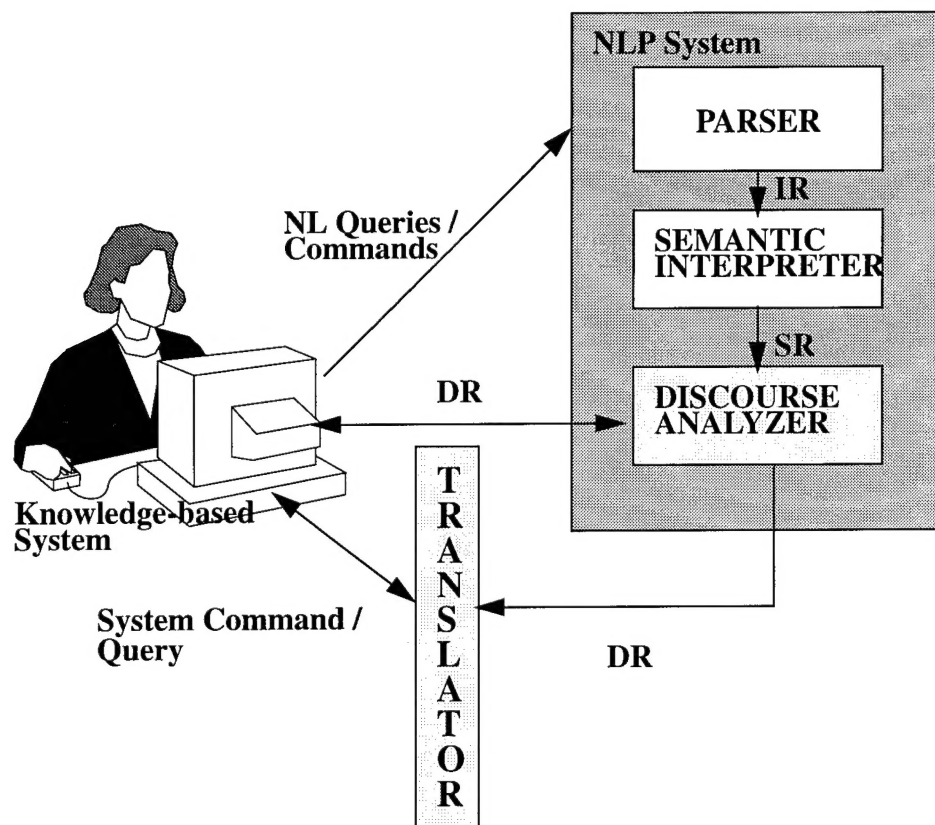


FIGURE 2. Natural Language Interface Architecture.

In this architecture, a sentence or set of sentences is input to a syntactic parser, which has access to a word dictionary and a grammar. The sentence may be a command to the expert system, update information for the expert system, or a query to the system. The parser translates the input sentence into one or more normalized forms (more than one if the sentence is ambiguous). The normalized form identifies operators, arguments, and modifiers in the sentence. This parse or intermediate representation (IR) is input to a semantic interpreter which executes lexical and thematic selectional restrictions and composes a data structure that represents the meaning of the natural language phrase at some level in some suitable formalism, typically logic- or frame-based (SR). Semantic interpretation can be executed either sequentially, after the parsing phase, or in tandem with parsing [Wauchope 1990]. The semantic representation (SR) is processed by a discourse analyzer which implements mechanisms for identifying, discourse elements, developing a model of the current discourse, and instantiating missing elements of the representation based on contextual information available from knowledge bases of the expert system or from pragmatic information available in the discourse analyzer. We assume some bi-directionality between the discourse analyzer and the knowledge-based system so that the discourse analyzer can take advantage of information in the expert systems knowledge bases and return to the discourse analyzer for further processing of incomplete information in the discourse representation (DR). Unlike the other components of the system, the discourse analyzer typically operates on the representation of sets of SRs of sentences rather than individual sentences. The DR of the dialog may be in some standard knowledge-representation language and become part of the knowledge base of the expert system, accessing information from the system, and/or adding or updating information to the knowledge base. If the DR is not compatible with the knowledge bases of the expert system, it is translated by the Translator either into some representation language the expert system understands or into expert system commands (System Command/Query).

CONCLUSIONS

We have suggested that, just as developers of natural language interface systems exploit discourse for cohesive, natural interface interactions, expert system developers should also exploit the dis-

course structure of the graphical dialog to provide friendlier expert system interfaces. By incorporating discourse modeling techniques into expert systems, systems will become easier to use, less redundant and repetitive, more natural and less opaque. In this paper, we have raised some issues that should be addressed in future research into expert system technology. In our research at the Naval Research Laboratory, we are addressing the following questions:

- Can discourse modeling contribute to the design of man-machine interfaces?
- How can discourse modeling be extended to non-natural language interfaces?
- Can discourse modeling in interfaces improve human performance on man-machine interfaces?

We propose to make discourse modeling a research concern today for next generation expert systems interfaces.

ACKNOWLEDGMENTS

This work was supported by the Office of Naval Research under project N00014-91-WX-24022, Program Element 61153N and by the Office of Naval Technology under project N00014-91-WX-4B839, Program Element 62234N. Tom Mifflin and Jim Kilgore provided the viewgraphs from which examples in Table 1 were drawn. The author thanks Dennis Perzanowski for his critical comments on this paper and Kenneth Wauchope for his technical insight. The notion that natural language technology might be of some use for other types of human-computer interfaces was developed as a result of many profitable discussions over the last year with Rob Carter, Rob Jacob, Preston Mullen, Astrid Schmidt-Nielsen, and Randy Shumaker.

REFERENCES

- Allen, J. (1984). Towards a general theory of action and time. *Artificial Intelligence*, 23(2), 123-154.
- Allen, J. (1987). *Natural language understanding*. Menlo Park, CA: Benjamin/Cummings Publishing Co., Inc.
- Carberry, S. (1988). Modeling the user's plans and goals. *Computational Linguistics* 14 (3), 23-37.
- Carberry, S. (1989). A pragmatics-based approach to ellipsis resolution. *Computational Linguistics* 15 (2), 75-96.
- Fox, B. (1990, March). Viewgraphs. Presentation given at ONR Cognitive Science Workshop, Princeton, NJ.
- Grosz, B.J. and C. Sidner. (1986). Attention, intention, and the structure of discourse. *Computational Linguistics* 12 (3).
- Hirst, G. (1987). *Semantic interpretation and the resolution of ambiguity*. New York, NY: Cambridge Univ. Press.
- Paap, K. and R. Roske-Hofstrand. (1988). Design of menus. M. Helander (ed.), *Handbook of human-computer interaction* (pp. 205-235). New York, NY: Elsevier.
- Perry, T.S. and J. Voelcker. 1989. Of mice and menus: designing the user-friendly interface. *IEEE Spectrum* (September), 46-51.
- Perzanowski, D. and B. Potter. (1991). InterFIS: a natural language interface to an expert system shell. In this volume.
- Reichman, R. (1985). *Getting computers to talk like you and me*. Cambridge, MA: MIT Press
- Scha, R. (1988). *Natural Language Interface Systems*. M. Helander (ed.), *Handbook of human-computer interaction* (pp. 941-956). New York, NY: Elsevier.
- Sidner, C. (1983). Focusing in the comprehension of definite anaphora. M. Brady and R.C. Berwick (eds.), *Computational models of discourse* (pp. 267-330). Cambridge, MA: MIT Press.
- Wauchope, K. (1990). A tandem semantic interpreter for incremental parse selection. *NRL Report 9288* (September 28, 1990).

Title: **Towards Friendlier User Interfaces for Expert Systems**

Author: Elaine Marsh

Keywords:

- discourse processing
- expert systems
- graphical interfaces
- human-computer interaction
- man-machine interfaces
- natural language interfaces
- natural language understanding